Use case of Grouper and webpage access

This is a beginner (newbie) example of using Grouper to restrict webpage access with php and ldap.

Use case

There is a small website of an index page, which has links to subpages. Certain sections should only be seen by certain populations of the institution; either the part of the index page, or the underlying deep linked webpage.

Here is a blurred version of the index page, and a description of who has access to what:

Note: the site is written in PHP

Grouper setup

There are three levels of access. Using permissions might be better, but since the site is PHP, and we only have groups and memberships in LDAP at Penn (not permissions), and we already have example code of PHP->LDAP, then we will just use group memberships to represent that access.

- Create a new folder for the application: site:apps:secureWebApp
- Create the adhoc group: site:apps:secureWebApp:adHocGroup (allow the client to edit the memberships (READ and UPDATE privileges), and give them the deep link URL to the lite membership UI:
- Create the three groups for the access levels, and add members of the other groups in Grouper that can be reused. Note, we add three new groups here to loosely couple and make things for flexible later on down the line:
  - site:apps:secureWebApp:facultyStaffStudents (add as member) site:community:facultyStaffStudents
  - site:apps:secureWebApp:facultyStaff (add as member) site:community:facultyStaff
  - site:apps:secureWebApp:orgAndAdHoc (add as members) site:community:employees:orgs:dept:whatever:org123 AND site:apps:secureWebApp:adHocGroup
- Our institution does WS authentication with kerberos principals and HTTP basic auth
  - Create the kerberos principal: secureWebApp_grouper/sever.site.edu
  - Add that kerberos principal to the kerberos subject table so it can be referenced as a subject from Grouper
  - Grant that kerberos principal access to LDAP (by adding to group: site:etc:ldapUsers), and grant READ to the three security groups: site:apps:secureWebApp:facultyStaffStudents, site:apps:secureWebApp:facultyStaff, site:apps:secureWebApp:orgAndAdHoc. At our institution, our LDAP is secure, so that allows that Kerberos principal to READ the memberships in LDAP

PHP code

We need to check which of the groups the user is in. We do not want to check this often, so we will store in PHP session, and if it is there, use it, if not, create it. Have an include for each secure page:

grouperSession.php

```
<?php

include 'grouper.php';

session_start();

//get this from SSO
$username = $_SERVER['REMOTE_USER'];

//not sure why this wouldnt be there
if (empty($username)) {
    echo "username not detected";
  exit( 1 );
}

//only allow backdoor for certain people, allow backdoor of ?backdoorUser=netid, TODO remove this when tested :)
if ($username == 'abc1' || $username == 'abc2' || $username == 'abc3') {

    if (!empty($_GET["backdoorUser"])) {

        $username = $_GET["backdoorUser"];

    }

}

//there is nothing emploitable in this comment, though in prod it should be removed since less information is
more secure
echo "<!-- \n";

//cache this in session so we dont hammer ldap
if (!isset($_SESSION['username']) || ($_SESSION['username'] != $username)  ) {

    echo "checking grouper...\n";
  $_SESSION['facultyOrStaff'] = ldapGroupHasMember("site:apps:secureWebApp:facultyStaff", $username);
  $_SESSION['facultyOrStaffOrStudent'] = ldapGroupHasMember("site:apps:secureWebApp:facultyStaffStudents",
$username);
  $_SESSION['orgOrAdHoc'] = ldapGroupHasMember("site:apps:secureWebApp:orgAndAdHoc", $username);
  $_SESSION['username'] = $username;


} else {

    echo "not checking grouper, using cache...\n";

}

echo "username: " . substr($username, 0, 1) . "...\n";
echo "facultyOrStaff: " . $_SESSION['facultyOrStaff'] . "\n";
echo "facultyOrStaffOrStudent: " . $_SESSION['facultyOrStaffOrStudent'] . "\n";
echo "orgOrAdHoc: " . $_SESSION['orgOrAdHoc'] . "\n";

echo "-->";
?>
```

grouper.php has the ldapGroupHasMember method:

```
<?php

//call e.g. if (ldapGroupLookup(
```

```
function ldapGroupHasMember($group, $username) {

  if (empty($group)) {

    throw new Exception('group is empty!');

  }

  if (empty($username)) {

    throw new Exception('username is empty!');

  }

  $filter = "(&(cn=" . $group . ")(hasMember=" . $username . ")(objectClass=*))";
  $pg_server = 'ldapserver.site.edu';
  $bind_dn = 'uid=secureWebApp_grouper/sever.site.edu,ou=users,dc=site,dc=edu';
  $base_dn = 'ou=ourgroups,dc=site,dc=edu';

  //find a group with this cn, that has hasMember of the username of the person logged in
  $attrs = array( 'cn' );
  include 'grouperPass.php';

  $lh = ldap_connect( $pg_server );
  if ( !$lh ) {
    echo "ldap_connect failed: " . ldap_error( $lh );
    exit( 1 );
  }
  ldap_set_option( $lh, LDAP_OPT_PROTOCOL_VERSION, 3 );
  ldap_set_option( $lh, LDAP_OPT_REFERRALS, 0 );
  if ( !ldap_start_tls( $lh )) {
    echo "ldap_start_tls failed: " . ldap_error( $lh );
    exit( 1 );
  }

  if ( !ldap_bind( $lh, $bind_dn, $pass)) {
    echo "ldap_bind failed: "  . ldap_error( $lh );
    exit( 1 );
  }

  $results = ldap_search( $lh, $base_dn, $filter, $attrs );
  if ( !$results ) {
    echo "ldap_search failed: " . ldap_error( $lh );
    exit( 1 );
  }


  $entries = ldap_get_entries( $lh, $results );
  $result = false;

  if ( $entries ) {
      for ( $i = 0; $i < $entries[ 'count' ]; $i++ ) {
        $entry = $entries[ $i ];
        for ( $j = 0; $j < $entry[ 'count' ]; $j++ ) {
          if ( !isset( $entry[ 'cn' ][ $j ] )) {
            continue;
          }

          //found it
          if ( $entry[ 'cn' ][ $j ] == $group) {
            $result = true;
          }
        }
      }

  }

  // this closes the connection, too.
  ldap_unbind( $lh );
```

```
  return $result;
}


?>
```

grouperPass.php has the password for the kerberos principal...

```
<?php

$pass = "***";

?>
```

Now, in each php page, we can include the grouperSession page near the top where an HTML comment is ok:

```
<?php
   include("grouperSession.php");
?>
```

For the subpages, we can see if the user is in the right group, depending on the page, or else see an error message. Note: the only reason someone would get the error is if an authorized user sent them a deep link (unless it is linked incorrectly)

```
<?php
   if (!$_SESSION['facultyOrStaff']) {

      echo "<br /><br /><h1>You are not allowed to access this page!</h1><br/><br />";

   } else {
?>
    HTML secure content here

<?php
   }
?>
```

For the index page, get all the sections, add them to an array if the user is allowed, and display them without empty slots. Note, give a message if the user is not allowed to see any:

```
<table width="100%">

          <?php

          class SiteSection {
            // property declaration
            public $imagePart;
            public $linkDescriptionPart;
          }


          $sections = array();

          $siteInformation = new SiteSection();
          $siteInformation->imagePart = '<img src="..." />';
          $siteInformation->linkDescriptionPart = '<h2 style="white-space: nowrap"><a href="anotherPage.php"
>Site Information</a></h2>'
             . '<p>Lorem Ipsum etc</p>';

          if ($_SESSION['facultyOrStaff']) {
            $sections[] = $siteInformation;
          }

          $projectInformation = new SiteSection();
          $projectInformation->imagePart = '<img src="..." />';
          $projectInformation->linkDescriptionPart = '<h2 style="white-space: nowrap"><a href="somePage.php"
>Some Information</a></h2>'
                . '<p>Lorem Ipsum etc</p>';

          if ($_SESSION['facultyOrStaffOrStudent']) {
            $sections[] = $projectInformation;
          }

          ...more sections...


          echo "<tr>\n";

          if (count($sections) == 0) {

              echo "<td width='100%'><h1>You need to be a Site student, faculty, or staff member to view
content on this page</h1></td>";

          }

          for ($i = 0; $i <= count($sections); $i++) {

              if (($i != 0) && ($i % 2 == 0)) {
              echo "</tr><tr>\n";
              }

              echo '<td width="96">' . $sections[$i]->imagePart . "</td>\n";
            echo '<td width="50%">' . $sections[$i]->linkDescriptionPart . "</td>\n";

          }

          echo "</tr>\n";

          ?>

        </table>
```

sdf